Deep Learning of Graph Matching

Andrei Zanfir2 Cristian Sminchisescu1,3
1Department of Mathematics, Faculty of Engineering, Lund University; 2Institute of Mathematics of the Romanian Academy; 3Google Research

Problem description
- Matching graph-based image representations, in order to find establish correspondences.
- The problem is formulated as a graph matching problem, with unary and pair-wise constraints.

\[ G_1 = (V_1, E_1) \quad G_2 = (V_2, E_2) \]

\[ |V_1| = n, |V_2| = m, |E_1| = p, |E_2| = q \]

Contributions
- We represent the unary and pair-wise structures as feature hierarchies with trainable parameters.
- We build novel deep network layers, associated with the problem of graph matching, that compute derivatives in an efficient way.
- We use the leading eigenvector in a voting based loss function, in order to learn correspondences.

Graph Matching
- Objective: \( v^* = \arg \max_{v} \text{Tr}(V^TMv) \) subject to \( I \cdot v \in \{0, 1\}^{n \times m} \)
- Relaxation: \( \|v\|_2 = 1 \)
- Solution: \( v^* \) is the leading eigenvector of \( M \)

Affinity Matrix Layer
- Affinity matrix factorization:
  \[ M = [\text{vec}(M_1)] + (G_1 \otimes G_2)[\text{vec}(M_2)](H_1 \otimes H_2)^T \]
- Simple solution to building the similarity matrices:
  \[ M_u = U^T U, M_x = X^T X \]
  \[ X_l = [F_l^T, F_l^T], X_r = [F_r^T, F_r^T] \]

Power Iteration Layer
- Forward pass: \( v_{k+1} = \frac{Mv_k}{\|Mv_k\|} \)
- 1D convolution: \( v_{k+1} = \frac{Mv_k}{\|Mv_k\|} \)
- Bi-stochastic Layer
  - We interpret \( v^* \) as an \( n \times m \) matrix, and make it double-stochastic:
    \[ v_{a} \sum \psi_{a} = 1; \sum \psi_{a} = 1 \]
  - We use matrix-backpropagation techniques to compute the gradients

Correspondence Loss
- Without exploiting the factorization of the affinity matrix:
  \[ \frac{\partial L}{\partial M} = M \cdot (I - v_k v_k^T) \frac{\partial L}{\|Mv_k\|} \]
  \[ \frac{\partial L}{\partial v_k} = M \cdot (I - v_k v_k^T) \frac{\partial L}{\|Mv_k\|} \]
- The complexities are \( O(m^3 n^2) \) and \( \Theta(m^2 n^2) \)

- Exploiting the factorization of the affinity matrix:
  \[ \frac{\partial L}{\partial M} = \sum_{i} G_i \cdot (I - v_{ki} v_{ki}^T) \frac{\partial L}{\|Mv_{ki}\|} \]
  \[ \frac{\partial L}{\partial v_k} = \sum_{i} G_i \cdot (I - v_{ki} v_{ki}^T) \frac{\partial L}{\|Mv_{ki}\|} \]
- The complexities are now \( O(\max(m^2 n^2, n^2 p)) \) and \( \Theta(pq + nm) \)

Results
- Sintel Dataset: \( n = m = 1024, M \) is of size \( 6 \times 1 \)
- CUB-200-2011 Dataset: \( 1 \) voxel per node

PASCAL VOC Keypoints